AMENDMENTS TO THE CLAIMS

This listing of claims replaces all prior versions, and listings, of claims in the application:

Listing of Claims:

1.   (Currently Amended) A computer-implemented attributed debugging system comprising:

a processor;

system memory;

a debugger that facilitates debugging of a computer software application, the debugger ~~obtains~~ obtaining from a computer user one or more attributes for attributed debugging, the attributes comprising keyword-like tags provided by the computer user which are configured to annotate various programming elements to indicate whether the computer user desires the elements to be displayed and, if so, how the computer user desires the elements to be displayed during debugging, the attributes including values of one or more properties of an object of the computer software application, the debugger being configured to allow the computer user to specify for each runtime object which object information is to be displayed and how that object information is to be displayed using attributes;

an expression evaluator, associated with the debugger, that examines a display proxy in place of the object, the display proxy is implemented as a private nested class of the object, the display proxy is configured to expose a subset of the one or more properties of the object, the subset excludes implementation-specific properties of the object; and

a variable display component that presents, to ~~a developer~~ the computer user, those specified values associated with the subset of the one or more properties of the object exposed by the display proxy, wherein only those values specified for display by the computer user are displayed and wherein the specified values are displayed in the manner indicated by the computer user.

2.   (Previously presented) The system of claim 1, the expression evaluator evaluates an expression associated with the object to determine the values, the expression is implemented in a particular programming language.

3.   (Previously presented) The system of claim 2, the programming language comprising at least one of C#, J# or Visual Basic.Net.

4.   (Previously presented) The system of claim 2, further comprising a plurality of expression evaluators, wherein each expression evaluator is associated with a different programming language.

5.   (Previously presented) The system of claim 1, the object comprises a class that includes at least one of a property or a method.

6.   (Previously presented) The system of claim 1, the expression evaluator creates an instance of the display proxy associated with the object of the computer software application.

7.   (Cancelled)

8.   (Previously presented) The system of claim 1, the display proxy has access to private implementation specifics of the object.

9.   (Previously presented) The system of claim 1, further comprising an attribute cache directory that stores an attribute associated with the display proxy, the expression evaluator employs the stored attribute to determine the values of the one or more properties of the object.

10.   (Previously presented) The system of claim 1, further comprising an editing component that facilitates modifying a value associated with the object.

11.   (Previously presented) The system of claim 1, the variable display employs at least one attribute associated with the object that provides a format to display the determined values of one or more properties of the object.

12.   (Previously presented) The system of claim 11, the attribute specifies the property is displayed.

13.   (Previously presented) The system of claim 12, the attribute employs an enumeration to specify the format of the display.

14.   (Currently Amended) The system of claim 13, the enumeration includes one enumeration value that indicates the property should not be displayed to the ~~developer~~ computer user.

15.   (Previously presented) The system of claim 13, the enumeration includes one enumeration value that indicates a hierarchical property is expanded by default.

16.   (Previously presented) The system of claim 13, the enumeration includes one enumeration value that indicates a hierarchical property is not expanded by default.

17.   (Previously presented) The system of claim 13, the enumeration includes one enumeration value that indicates a hierarchical property itself is not displayed and members of the hierarchical property are displayed.

18.   (Previously presented) The system of claim 11, the attribute specifies what is displayed for a class.

19.   (Previously presented) The system of claim 18, the attribute includes an argument that comprises a string that is displayed in a value column for an instance of the class.

20.　　(Previously presented) The system of claim 18, the argument is associated with a property of the class.

21.   (Currently Amended) A method facilitating attributed debugging comprising:

receiving a request to examine details of one or more properties of an object in a computer software application being debugged;

receiving from a computer user one or more display proxy attributes for attributed debugging, the display proxy attributes comprising keyword-like tags provided by the computer user which are configured to annotate various programming elements to indicate whether the computer user desires the elements to be displayed and, if so, how the computer user desires the elements to be displayed during debugging in the debugger, the display proxy attributes including values of one or more properties of an object of the computer software application, the debugger being configured to allow the computer user to specify for each runtime object which object information is to be displayed and how that object information is to be displayed using attributes;

~~determining whether a~~ accessing the received display proxy attribute ~~exists~~ for the object, the display proxy is implemented as a private nested class of the object such that the display proxy within the definition of the object, the display proxy provides relevant properties regarding a state of the object and conceals properties related to implementation of the object;

creating a display proxy for the object in accordance with the display proxy attribute; ~~and~~

examining the display proxy in place of the object to determine debug information related to the object;

presenting to the computer user the object associated with the display proxy, wherein only those objects specified for display by the computer user are displayed and wherein the specified objects are displayed in the manner indicated by the computer user.

22.   (Currently Amended) The method of claim 21 further comprising providing the debug information determined by the display proxy to ~~a developer~~ computer user in response to the request to examine the object.

23.   (Original) A computer readable medium having stored thereon computer executable instructions for carrying out the method of claim 21.

24-25. (Cancelled)

26.   (Currently Amended) An attributed debugging system comprising:

processing means;

memory means;

means for receiving a request to debug an object in a computer software application;

means for receiving from a computer user one or more display proxy attributes for attributed debugging, the display proxy attributes comprising keyword-like tags provided by the computer user which are configured to annotate various programming elements to indicate whether the computer user desires the elements to be displayed and, if so, how the computer user desires the elements to be displayed during debugging in the debugger, the display proxy attributes including values of one or more properties of an object of the computer software application, the debugger being configured to allow the computer user to specify for each runtime object which object information is to be displayed and how that object information is to be displayed using attributes;

means for ~~determining whether a~~ accessing the received display proxy attribute ~~exists~~ in association with the object, the display proxy is implemented as a private nested class of the object such that the display proxy within the definition of the object, the display proxy provides relevant properties regarding a state of the object and conceals properties related to implementation of the object;

means for creating an instance of the display proxy;

means for examining the display proxy in place of the object, means for examining the display proxy comprise means for obtaining values of the provided relevant properties; ~~and~~

means for providing debug information in response to the request to debug the object, the debug information is based at least in part on the display proxy and includes at least the obtained values of the provided relevant properties;

means for presenting to the computer user the object associated with the display proxy, wherein only those objects specified for display by the computer user are displayed and wherein the specified objects are displayed in the manner indicated by the computer user.

27.   (Currently Amended) A computer-implemented attributed debugging system, comprising:

>   a processor;
>
>   system memory;
>
>   a debugger that facilitates debugging of a computer software application, the debugger obtaining from a computer user one or more attributes for attributed debugging, the attributes comprising keyword-like tags provided by the computer user which are configured to annotate various programming elements to indicate whether the computer user desires the elements to be displayed and, if so, how the computer user desires the elements to be displayed during debugging, the attributes including values of one or more properties of an object of the computer software application, the debugger being configured to allow the computer user to specify for each runtime object which object information is to be displayed and how that object information is to be displayed using attributes,
>
>   wherein the attributes are configured to control the following based on the computer user's indication:
>
>>   which value is to be shown at the top-level for a given type whereby the need to expand the object for additional information is reduced;
>>
>>   whether or not a field or property for a programming element should be shown or not shown;
>>
>>   what the value should be shown for a field or property of the programming element;
>>
>>   whether a type is to be shown fully expanded; and
>>
>>   adding descriptive name or type information to the programming element;
>
>   an expression evaluator, associated with the debugger, that examines a display proxy in place of the object, the display proxy is implemented as a private nested class of the object, the display proxy is configured to expose a subset of the one or more properties of the object, the subset excludes implementation-specific properties of the object; and

a variable display component that presents, to the computer user, those specified values associated with the subset of the one or more properties of the object exposed by the display proxy, wherein only those values specified for display by the computer user are displayed and wherein the specified values are displayed in the manner indicated by the computer user.

~~a debugger that facilitates debugging a computer software application;~~

~~an expression evaluator associated with the debugger that examines at least one object of the computer software application, the expression evaluator determines debug information comprising states of the at least one object, the states include values for at least one property of the at least one object;~~

~~the expression evaluator inspects the at least one object to verify if each of the at least one object includes a display proxy defined as a private nested class of the object, the display proxy provides relevant properties of an associated object and conceals properties related to implementation of the object, the expression evaluator examines the display proxy in place of the object to determine debug information that includes values for the relevant properties of the at least one object;~~

~~an attribute cache directory that retains instances of one or more display proxies, the expression evaluator queries the attribute cache directory for instances of display proxies associated with the at least one object, the expression evaluator creates an instance if not retained in the attribute cache directory; and~~

~~a variable display component that presents the debug information to a developer computer user, the debug information includes values of relevant properties of the at least one object determined from examination of display proxies of the at least one object.~~